Collective Principal Component Analysis from Distributed, Heterogeneous Data

Hillol Kargupta, Weiyun Huang, Krishnamoorthy Sivakumar, Byung-Hoon Park, and Shuren Wang

School of Electrical Engineering and Computer Science
Washington State University
Pullman, WA 99164-2752, USA
{hillol, whuang1, siva, bhpark, swang}@eecs.wsu.edu

Abstract. Principal component analysis (PCA) is a statistical technique to identify the dependency structure of multivariate stochastic observations. PCA is frequently used in data mining applications. This paper considers PCA in the context of the emerging network-based computing environments. It offers a technique to perform PCA from distributed and heterogeneous data sets with relatively small communication overhead. The technique is evaluated against different data sets, including a data set for a web mining application. This approach is likely to facilitate the development of distributed clustering, associative link analysis, and other heterogeneous data mining applications that frequently use PCA.

1 Introduction

Representation plays an important role in data analysis and modeling. Principal component analysis (PCA) offers a technique to construct an adaptive representation of the data that exploits its underlying distribution. PCA detects the dependencies among the features defining the data and constructs a number of basis vectors (often fewer than the number of features) that can be used to represent the data in a compact and efficient manner. PCA is frequently used in data mining applications [1,2].

PCA involves computation of the dominant eigenvectors of the covariance matrix generated by the data. There exist several well-known efficient techniques for performing PCA [3,4]. Most of the established techniques for PCA require the data set to be centrally located. However, the emergence of network-based computation has offered a new challenge to the traditional practice of PCA. This has introduced a new important dimension to the PCA problem — distributed sources of data. Performing PCA from distributed data tables is of growing importance and it is still an open question. To the best of our knowledge, there does not exist any technique that can perform distributed PCA from heterogeneous data sites (i.e. sites with different tables storing data for different features) with minimal data communication.

This paper offers one solution to this problem—the Collective PCA. It contributes a new technique to the growing body of the Distributed Knowledge Discovery (DKD) [5,6,7] literature.

Section 2 presents the Collective PCA (CPCA) algorithm. Section 3 presents the experimental results. Section 4 concludes this paper.

2 Collective PCA

Let us first consider the CPCA algorithm from an abstract algorithmic perspective. Let X be the entire relational data set and it is stored in different tables at different sites. In this paper we shall consider heterogeneous data sites [7] where different sites observe different sets of features. In particular, each site has data regarding a particular subset of the n features. For the sake of simplicity we shall assume that each site has only one table. Let us assume that there are s sites and X can be partitioned as $X = [X_1, X_2, \dots, X_s]$, where X_i is a $m \times n_i$ submatrix of X that is available at site i . A PCA of this distributed data in a centralized fashion would involve moving data to one central site and calculating the eigenvalues/eigenvectors of the covariance matrix $X'X$ of the global data matrix X . The amount of data to be moved is $O(mn)$, where m is the number of data samples (rows of the global data table X) and n is the total number of features (columns of the global data table X). For typically DDM-applications this amount of data communication is either prohibitive because of the limited bandwidth or impractical because of logistics and/or security related reasons.

In the CPCA approach, we perform the computations for PCA (to the extent possible) locally, thereby minimizing the amount of data communication and the computation at the central site. First we perform a local PCA on the data partition X_i at site i . Let A_i be the $n_i \times k_i$ matrix whose columns are the k_i eigenvectors corresponding to the k_i largest eigenvalues of $X_i'X_i$. Let

$$Y_i = X_i A_i \quad (1)$$

be the principal components computed at site i . The choice of k_i will depend on the eigenvalues of the local covariance $X_i'X_i$ and would be dictated by our error tolerance. It is a tradeoff between dimensionality reduction and accuracy.

These principal components can be used to obtain a reasonable approximation to the original data as follows:

$$\hat{X}_i = Y_i A_i' \quad (2)$$

The relative mean-squared error (RMSE) between X_i and $\hat{X}_i$ can be expressed in terms of the eigenvalues of the covariance matrix (Σ_x) as follows:

$$\frac{E[(X_i - \hat{X}_i)(X_i - \hat{X}_i)']}{E[X_i X_i']} = \frac{\sum_{j=k+1}^n \lambda_j}{\sum_{j=1}^n \lambda_j}. \quad (3)$$

If the eigenvalues of Σ_x are “spread-out” in the sense that $\lambda_{\max}/\lambda_{\min}$ is large, the data X maybe represented (with a small RMSE) by a small number of principal components.

In all our experiments, we chose k_i to yield a RMSE of 0.1 (see eq. (3)). Since the number of rows in large tables (m) is very large, we select a subset of

c samples (rows) (where $c \ll m$), with uniform probability. With some abuse of notation, we will denote by Y_i the $c \times k_i$ matrix consisting of the principal components and only c selected samples. The individual Y_i and A_i from each site are then transmitted to a central site. At the central site, a new $c \times k$ data matrix Y is formed by putting together the data from the individual sites $Y = [Y_1, Y_2, \dots, Y_s]$, where $k = \sum_{i=1}^s k_i$. The data communication involved here is $O(ck)$ for the Y_i 's and $O(\sum_i n_i k_i)$ for the A_i 's (compare with $O(mn)$ for the centralized PCA case, where $c \ll m$ and $k = \sum_{i=1}^s k_i \ll n$). Typically, the number of samples c that are selected is much larger than the number of features n_i at site i . Therefore, the overhead involved in transmitting A_i would be negligible compared to the overall data transmission involved.

At the central site, in principle, we need to reconstruct the original data X from the Y_i 's and A_i 's (see eq. (2)) and perform a PCA based on this reconstructed data. It follows from eq. (1) that Y is a linear transformation of X with $Y = XA$, where A is a block diagonal matrix with matrices $A_1, \dots, A_s$ on the diagonal. Since the PCA is invariant to linear transformations [8], we can work with the Y data instead of X . The primary advantage is that we can exploit the dimensionality reduction already achieved at each of the local sites. Let v_i , $i = 1, 2, \dots, p$ be the eigenvectors, corresponding to the p largest eigenvalues of the covariance matrix of Y . Then $w_i = Av_i$ are the required eigenvectors of the covariance $X'X$ of the original data X . The following section presents the experimental results.

3 Experiments with the CPCA

This section presents results of applying the CPCA to two experimental test suites. The first experimental suite tests the scalability of CPCA to large data sets. The second suite considers the application of CPCA in a real-life application scenario.

3.1 Experiment Suite: I

This section tests the overall performance including the scalability of the CPCA. We used the Quest Synthetic Data Generation Code ¹ to generate a data set with 60,000 observations and 200 features. Although this code generates labeled data, we did not use the class-label for the unsupervised CPCA approach.

We partitioned the data set into two subsets, each containing 100 features. We select the set of PCs that gives a RMSE of 0.1. Figure 1(Left) shows two curves, each representing the variation of the angle (in radian) between a CPCA-generated dominant principal component and the corresponding principal direction generated from the centralized data. Note that the angle between two vectors represents the distance between them and therefore it is a good measure of the accuracy of the estimated PCs.

¹ <http://www.almaden.ibm.com/cs/quest/syndata.html>

Table 1. Size of data set, number of sites, and number of selected local principal components.

Data	No. of rows	No. of features	No. of sites	Total number of selected local PCs
Quest	60000	200	2	161
Quest	60000	200	10	164

Figure 1(Right) shows similar performance when the data set is distributed among ten sites. Table 1 shows the size of the global data set and the total number of selected local principal components from all the sites.

We see from the figures that the CPCA achieves good compression. Suppose our global data set is m by n , and is distributed among s sites, such that each site contains n_i features. Number of selected local principal component is k_i , and the total number of selected local principal components is $k = \sum_{i=1}^s k_i$. Usually k is quite small compared to n . Let us denote the number of sampled rows by c ; for our experiments we chose a value of c such that $c < 0.2 * m$. The total communication cost is $O(c * k + \sum_{i=1}^s n_i * k_i)$, which is very small compared to $O(m * n)$, the cost to move the whole data set to one site.

3.2 Experiment Suite: II

In this section we illustrate the technique in the context of a practical application scenario. PCA is frequently used for high-dimensional text-analysis applications. Therefore text analysis should be an ideal candidate for applying the CPCA. However, in a distributed environment, applications may become more interesting and challenging when relevant data sets involve text, numeric, and other non-numeric features. In the following we describe one such case.

Consider the financial news stories regularly posted on the Internet. These news stories are often very useful for investors, portfolio managers, and others.

Fig. 1. (Left) Performance of CPCA for the Quest Synthetic data with two data sites. (Right) Performance of CPCA for the Quest Synthetic data with ten data sites.

Fig. 2. (Left)Performance of CPCA for the Company Profile data with 3 data sites. (Right) Dimension compression in case of the Company Profile data.

Typically many of these stories are associated with some companies. Announcements regarding new products, quarterly revenue, legal battles, mergers, and partnerships of companies often dominate such business news. On the other hand, there exist many sites in the Internet that offer valuable information about the background and current financial profile about almost all major companies. For example Yahoo finance, CNN finance are only a few among many excellent sources for such information. The web sites of the company itself also provides quite useful information such as new developments, products, and others.

The experiment performed in this section considers a real-life applications using three data tables, collected using the information available over the Internet. One table corresponds to news corpora. Each document in this corpora corresponds to a key. For our application we used a unique code associated with every company name as the key. A second table is constructed that stores the financial and background feature values of a company, again indexed by its company symbol. A third data table is constructed using the data about the sector the company belongs to. This table is also indexed by the company name. We considered a data set involving 1,027 companies and the three tables are located at three different sites. The CPCA technique is applied on this three data sets. The RMSE threshold is set to 0.1. Table 2(Right) shows the number of features at each site and corresponding number of chosen PCs. As we see, the CPCA technique offers a big compression factor. Figure 2(Left) shows two curves, each representing the variation of the angle between a CPCA-generated dominant principal component and the corresponding principal direction generated by the centralized approach. The following section concludes this paper.

4 Conclusions

Distributed data analysis is playing an increasingly important role in knowledge discovery from multiple databases distributed over a network. The growing popu-

larity of environments with “thin” computing devices (e.g. wearable computers, palmtops, and laptops) connected to powerful servers through limited bandwidth (possibly wireless) networks is aiding the rapid development of distributed data mining techniques that do not require communication of large amount of data. PCA is frequently used for high dimensional data mining applications. Development of distributed PCA algorithm is therefore important. This paper presented the collective PCA technique which offers one solution to this problem. The experimental results demonstrated that the CPCA can be effectively used for analyzing high dimensional data of large quantity.

We are in the process of applying the CPCA technique for distributed clustering of heterogeneous data. We are also working towards bounding the error introduced by the approximations used by the CPCA. We believe that this simple but effective technique will be quite useful in further advancing the practice of the DKD technology.

Acknowledgement

The authors thank American Cancer Society for supporting part of this research.

References

1. Boley, D.: Principal direction divisive partitioning. *Data Mining and Knowledge Discovery* **2** (1998) 325–344
2. Faloutsos, C., Korn, F., Labrinidis, A., Kotidis, Y., Kaplunovich, A., Perkovic, D.: Quantifiable data mining using principal component analysis. Technical report (1997) Institute for Systems Research, University of Maryland technical Report TR 97-25.
3. Golub, G.H., Loan, C.F.V.: *Matrix Computations*. The Johns Hopkins University Press, Baltimore (1989)
4. Anderson, E., Bai, Z., Bischof, C., Blackford, S., Demmel, J., Dongarra, J., Croz, J.D., Greenbaum, A., Hammarling, S., McKenney, A., Sorensen, D.: *LAPACK Users' Guide*. Third edn. Society for Industrial & Applied Mathematics (1999)
5. Chan, P.K., Stolfo, S.J.: Sharing learned models among remote database partitions by local meta-learning. In Simoudis, E., Han, J., Fayyad, U., eds.: *The Second International Conference on Knowledge Discovery and Data Mining*, AAAI Press (1996) 2–7
6. Grossman, R., Bailey, S., Kasif, S., Mon, D., Ramu, A., Malhi, B.: The preliminary design of papyrus: A system for high performance, distributed data mining over clusters, meta-clusters and super-clusters. *Fourth International Conference of Knowledge Discovery and Data Mining*, New York, New York, Pages 37–43 (1998)
7. Kargupta, H., Park, B., Hershberegger, D., Johnson, E.: Collective data mining: A new perspective toward distributed data mining. To be published in the *Advances in Distributed and Parallel Knowledge Discovery*, Eds: Hillol Kargupta and Philip Chan, AAAI/MIT Press (1999)
8. Jackson, J.E.: *A User's Guide to Principal Components*. John Wiley (1991)